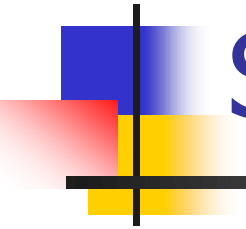


Software Engineering 3

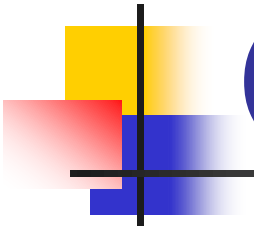
Lecture 4 – BDD

Eng. Sally Jarkas
Eng Aya Joumaa

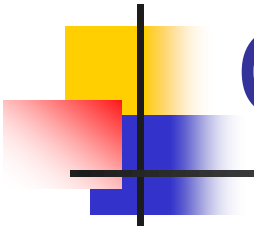
Behavior-Driven Development (BDD)

- BDD is a design activity where you build pieces of functionality incrementally guided by the expected behavior.
- BDD is a way for software teams to work that closes the gap between business people and technical people by:
  - Encouraging collaboration across roles to build shared understanding of the problem to be solved
  - Working in rapid, small iterations to increase feedback and the flow of value
  - Producing system documentation that is automatically checked against the system's behaviour

Behavior-Driven Development (BDD)



- we use the BDD format to ensure the syntax detailing the requirements, is close to the syntax an engineer would use to write and execute tests.



Gherkin

- Originally intended for developers, Gherkin is a structured approach to writing (BDD).
- Instead of testing little bits of code, behavioral tests focus on how users interact with your system.
- Gherkin uses a set of special keywords to give structure and meaning to executable specifications.

Gherkin statements

- Gherkin is a Domain Specific Language for writing acceptance criteria that has five main statements:
  - Scenario — a label for the behavior you're going to describe
  - Given — the beginning state of the scenario
  - When — a specific action that the user takes
  - Then — a testable outcome, usually caused by the action in When
  - And — this continues any of the other three operators

Benefits of using Gherkin

- As a PM, the benefits of using Gherkin are:
  - You'll catch missing workflows before any work is started.
  - It's a direct correlation to the user workflows that you and Design have developed.
  - Developers know when the story is Done because they have clear acceptance criteria.
  - A consistent language across stories helps the team focus on delivering user value, unhindered by your writing style that day.

Gherkin – Example 1

Feature: Refund item

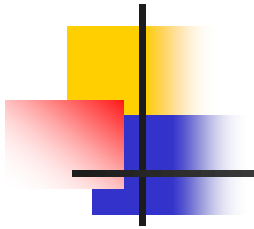
- **Scenario**: Jeff returns a faulty microwave
  Given Jeff has bought a microwave for $100
  And he has a receipt
  When he returns the microwave
  Then Jeff should be refunded $100

Gherkin – Example 2

- Feature: As a user I want to sign in so I can see my marketing campaigns
- **Scenario: User supplies correct user name and password**
  Given that I am on the sign-in page
  And I enter my user name and password correctly
  When click 'Sign In'
  Then I am taken to the dashboard

Gherkin – Example 3

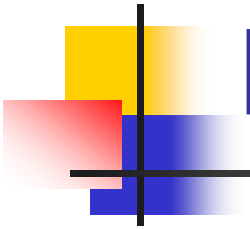
- **Scenario: User does NOT supply correct user name and password**
  Given that I am on the sign-in page
  And I enter my user name and password incorrectly
  When click 'Sign In'
  Then I see an error message 'Sorry, incorrect user name or password."



BDD Anti-Patterns (Mistakes)

BDD - Example 1

- Here's one misconception that is most commonly found:
- **Given** the Form is submitted
  When the value entered in the Number text box is not numerical
  Then an error message "Please enter a numerical value" appear
- it clearly violates the basic principle of the *Given-When-Then* format



BDD - Example 1

- better way to write the previous example
- **Given** the value entered in the Number text box is not numerical
  When the Form is submitted
  Then an error message "Please enter a numerical value" appear.

BDD - Example 2

- **Given** the User is logged in ← Condition
  And the value in the Number text box changes ← Trigger
  When the value in it is not numerical ← Condition? Trigger?
  Then an error message "Please enter a numerical value" appears
- it clearly violates the basic principle of the *Given-When-Then* format

BDD – Example 3

- Given I'm at the sign-up form
  When I enter my details

  First Name
  Last Name
  Email
  Password

  And I submit the Form

Then an account is created
And account name is set as my Email
And a confirmation email is sent to me

BDD – Example 3

Given I'm at the sign up form
And all these mandatory fields are entered

First Name

Last Name

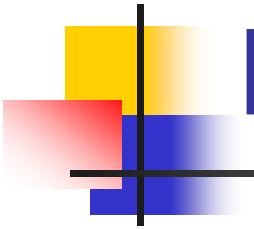
Email

Password

When I submit the Form
Then an account is created
And account name is set as my Email
And a confirmation email is sent to me

Precondition vs sequence of events

- **Given** I have selected a flight at the Flight Selection page
  And I chose to skip seat selection at the Seat Selection page
  And I selected a meal at the Meal Selection page
  When I confirm my details on the Confirm Details page
  Then a warning message should appear "Seat will be assigned randomly, proceed".
- Again, at first glance, this looks right, and frankly, it is not hard to write acceptance tests for this. Yet, there is a simpler, and better way of writing the same scenario



Example

- **Feature: Google Searching**

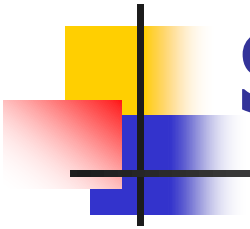
As a web surfer, I want to search Google, so that I can learn new things.

- Scenario: Simple Google search

  Given a web browser is on the Google page

  When the search phrase “Gherkin" is entered

  Then results for " Gherkin " are shown



Step Tables

Feature: Google Searching
As a web surfer, I want to search Google, so that I can learn new things.

Scenario: Simple Google search
Given a web browser is on the Google page
When the search phrase " Gherkin " is entered
Then results for " Gherkin " are shown
And the following related results are shown

related
Gherkin Express
Gherkin videos

The Background Section

Feature: Google Searching

As a web surfer, I want to search Google, so that I can learn new things.

Background:

Given a web browser is on the Google page

Scenario: Simple Google search for Samsung New Devices

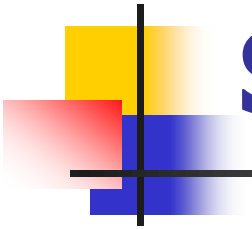
When the search phrase " Samsung New Devices " is entered

Then results for " Samsung New Devices " are shown

Scenario: Simple Google search for Laptops

When the search phrase " Laptops " is entered

Then results for " Laptops " are shown



Scenario Outline

- instead of “Scenario” and instead of getting input from steps, the input for test is in a DataTable under **Examples**.

Scenario Outline

Feature: Addition
Scenario Outline: Sum of two numbers
Given first number is <firstNumber>
And second number is <secondNumber>
When user executes sum function
Then the sum is <result>

Examples:

firstNumber	secondNumber	result
10	20	30
50	60	110

Lecture Example

- Feature: Account Holder withdraws cash

  Scenario: Account has sufficient funds

- Given the account balance is $100
  And the card is valid
  And the machine contains enough money
  When the Account Holder requests $20
  Then the ATM should dispense $20
  And the account balance should be $80
  And the card should be returned

- https://medium.com/@mvwi/story-writing-with-gherkin-and-cucumber-1878124c284c
- https://cucumber.io/docs/gherkin/
- https://cucumber.io/blog/bdd/user-stories-are-not-the-same-as-features/